

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) An integrated tracing and logging system employed within a network comprising:

a computer system having a processor coupled with a memory, the computer system further including,

a tracing module associated with specified program code regions of an application, the tracing module to receive via an application programming interface (API) and process tracing method calls generated by the application when the specified program code regions are executed[[;]],

a logging module associated with specified categories related to the network, the logging module to receive via the API and process logging method calls from network components associated with the categories[[;]], and

a formatter coupled to the tracing module and the logging module, the formatter to receive an output from one of the tracing module and the logging module, the formatter including a configuration file storing a format definition for the formatter, the configuration file of the formatter further to receive a change to the format definition for the formatter during a runtime of the integrated tracing and logging system, wherein receiving the change to the format definition for the formatter does not require recompiling of any source code of the integrated tracing and logging system, wherein the formatter to provide an indication of the format to the tracing module or the logging module via the API; wherein processing tracing method calls or processing logging method calls includes receiving from the formatter an indication of a format for a message to be sent from the respective tracing module or logging module, the respective tracing module or logging module further to format the message according to the indicated format the output from the

one of the tracing module and the logging module according to the changed format definition; and

an output destination to receive the formatted ~~message from~~ output of the ~~at least~~ one of the tracing module and the logging module.

2. (Original) The system of claim 1, wherein the formatter is one of a list formatter, a human-readable formatter, and a markup language formatter.

3. (Previously Presented) The system of claim 1, wherein one or more properties of the formatter are defined in the configuration file.

4. (Original) The system of claim 3, wherein the configuration file includes an identifier to identify the formatter.

5. (Original) The system of claim 3, wherein the one or more properties are formatted as key-value-pair properties, each key-value pair having a key to specify an attribute and a value to provide a definition for the specified attribute.

6. (Original) The system of claim 3, wherein the configuration file defines the message format for the received message, the message format including one or more fields.

7. (Original) The system of claim 6, wherein the one or more fields of the message format includes at least one of

a timestamp field to indicate a time for the received message;

a location of origin field to indicate a source of the received message;

a thread identifier field to indicate a thread associated with the received message;

a message severity indicator field to indicate a severity level of the received message;
and

a message identifier field to identify the received message.

8. (Original) The method of claim 1, wherein the output destination is at least one of
a trace file; and
a log file.

9. (Original) The method of claim 1, wherein the output destination is a console.

10. (Currently Amended) A computer-implemented method employed within a network
comprising:

creating an instance of a tracing controller associated with specified program code
regions of an application, the tracing controller instance to receive and process tracing
method calls generated by the application when the specified program code regions are
executed;

creating an instance of a logging controller associated with specified categories
related to the network, the logging controller to receive and process logging method calls
from network components associated with the categories;

providing a common application programming interface of the tracing controller
instance and the logging controller instance, whereby the tracing controller instance and the
logging controller instance are accessed; ~~specifying an output destination to receive via the
common application programming interface of the tracing controller instance and the logging
controller instance a message from at least one of the tracing controller instance and the
logging controller instance; and selecting a formatter to provide a message format for the
received message, wherein the message format is defined based, at least in part, on a
configuration file~~

creating an instance of a formatter coupled to the tracing module and the logging module, the formatter including a configuration file storing a format definition for the formatter;

after the creating the instance of the formatter, changing the format definition stored in the configuration file, wherein changing the format definition does not require a recompiling of any source code;

receiving at the formatter an output from one of the tracing module and the logging module; and

formatting at the formatter the output from the one of the tracing module and the logging module, the formatting according to the changed format definition of the configuration file.

11. (Original) The method of claim 10, further comprising:
configuring the message format for the selected formatter.
12. (Original) The method of claim 11, wherein configuring the message format comprises providing an identifier to the configuration file to identify the selected formatter.
13. (Original) The method of claim 12, wherein configuring the message format further comprises specifying one or more fields for the message format.
14. (Original) The method of claim 13, wherein specifying one or more fields comprises specifying at least one of
 - a timestamp field to indicate a time for the received message;
 - a location of origin field to indicate a source of the received message;
 - a thread identifier field to indicate a thread associated with the received message;

a message severity indicator field to indicate a severity level of the received message;
and

a message identifier field to identify the received message.

15. (Original) The method of claim 10, further comprising:
providing a filter to the specified output destination to selectively filter the message.

16. (Currently Amended) A system comprising:

a computer system having a processor coupled with a memory, the computer system further including,

a means for creating an instance of a tracing controller associated with specified program code regions of an application, the tracing controller instance to receive via an application programming interface (API) and process tracing method calls generated by the application when the specified program code regions are executed[[;]],

a means for creating an instance of a logging controller associated with specified categories related to the network, the logging controller to receive via the API and process logging method calls from network components associated with the categories,

means for creating an instance of a formatter to receive an output from one of the tracing controller instance and the logging controller instance, the formatter instance including a configuration file storing a format definition for the formatter instance, the formatter to provide an indication of the format to the tracing controller instance or the logging controller instance via the API; wherein processing tracing method calls or processing logging method calls includes determining from the provided indication a format for a message to be sent from the respective tracing controller instance or logging controller instance, the respective tracing controller instance or logging controller and

means for changing the format definition stored in the configuration file for the formatter instance, wherein the changing the format definition does not require a recompiling of any source code, wherein the formatter instance further to format the message output from the one of the tracing controller instance and the logging controller instance, the formatting according to the determined message changed format definition stored in the configuration file: and a means for specifying an output destination to receive the formatted message from output of the respective one of the tracing controller instance [] or [] and the logging controller instance [] ; [].

Claims 17. (Canceled).

18. (Currently Amended) The system of claim [] 17 [] 16, wherein the means for configuring the message format changing the format definition comprises:

a means for specifying one or more fields for [] the [] a defined message format.

19. (Currently Amended) The system of claim 18, wherein the means for specifying one or more fields comprises a means for specifying at least one of

a timestamp field to indicate a time for the received message output;

a location of origin field to indicate a source of the received message output;

a thread identifier field to indicate a thread associated with the received message output;

a message severity indicator field to indicate a severity level of the received message output; and

a message identifier field to identify the received message output.

20. (Currently Amended) An article of manufacture comprising:

an electronically accessible medium providing instructions that, when executed by an apparatus, cause the apparatus to

create an instance of a tracing controller associated with specified program code regions of an application, the tracing controller instance to receive and process tracing method calls generated by the application when the specified program code regions are executed;

create an instance of a logging controller associated with specified categories related to the network, the logging controller to receive and process logging method calls from network components associated with the categories;

provide a common application programming interface of the tracing controller instance and the logging controller instance, whereby the tracing controller instance and the logging controller instance are accessed; ~~specify an output destination to receive via the common application programming interface of the tracing controller instance and the logging controller instance a message from at least one of the tracing controller instance and the logging controller instance; and select a formatter to provide a message format for the received message, wherein the message format is defined based, at least in part, on a configuration file~~

create an instance of a formatter coupled to the tracing module and the logging module, the formatter including a configuration file storing a format definition for the formatter;

after the creating the instance of the formatter, change the format definition stored in the configuration file, wherein changing the format definition does not require a recompiling of any source code;

receive at the formatter an output from one of the tracing module and the logging module; and

format at the formatter the output from the one of the tracing module and the logging module, the formatting according to the changed format definition of the configuration file.

Claim 21. (Canceled).

22. (Currently Amended) The article of manufacture of claim 20, wherein the instructions that, when executed by the apparatus, cause the apparatus to ~~configure~~ change the message format definition for the ~~selected~~ formatter cause the apparatus to provide one or more fields for ~~[[the]]~~ a defined message format.

23. (Currently Amended) An apparatus comprising:

an application; and

a processor and logic executable thereon to

create an instance of a tracing controller associated with specified program code regions of the application, the tracing controller instance to receive and process tracing method calls generated by the application when the specified program code regions are executed;

create an instance of a logging controller associated with specified categories related to a network, the logging controller to receive and process logging method calls from network components associated with the categories;

provide a common application programming interface of the tracing controller instance and the logging controller instance, whereby the tracing controller instance and the logging controller instance are accessed; ~~specify an output destination to receive via the common application programming interface of the tracing controller instance and the logging controller instance a message from at least one of the tracing controller instance and the logging controller instance; and select a formatter to provide a message format for the received message, wherein the message format is defined based, at least in part, on a configuration file~~

create an instance of a formatter coupled to the tracing module and the logging module, the formatter including a configuration file storing a format definition for the formatter;

after the creating the instance of the formatter, change the format definition stored in the configuration file, wherein changing the format definition does not require a recompiling of any source code;

receive at the formatter a message from one of the tracing module and the logging module; and

format at the formatter the message from the one of the tracing module and the logging module, the formatting according to the changed format definition of the configuration file.

24. (Original) The apparatus of claim 23, wherein the selected formatter is one of a list formatter, a human-readable formatter, and a markup language formatter.

25. (Original) The apparatus of claim 23, wherein the configuration file includes an identifier to identify the formatter.

Claim 26. (Canceled).

27. (Currently Amended) The apparatus of claim ~~26~~ 23, wherein the processor and logic executable thereon to ~~configure~~ change the message format definition comprises a processor and logic executable thereon to specify one or more fields for ~~the~~ a defined message format.

28. (Original) The apparatus of claim 27, wherein the processor and logic executable thereon to specify one or more fields for the message format comprises a processor and logic executable thereon to specify at least one of

a timestamp field to indicate a time for the received message;

a location of origin field to indicate a source of the received message;
a thread identifier field to indicate a thread associated with the received message;
a message severity indicator field to indicate a severity level of the received message;
and
a message identifier field to identify the received message.